

- **ejercicio 1:** Dados dos hilos:  
A - > sysout("a"); sysout("b");  
B - > sysout("c"); sysout("d");  
Implementar un semáforo de manera que B no pueda imprimir en consola "d" hasta que A no haya impreso "a" en la misma.
- **ejercicio 2:** Demostrar que cuando un semáforo bloquea un hilo en java, el mismo pasa a estado *WAITING*.
- **ejercicio 3:** Implementar semáforos de manera que los hilos del ejercicio 1 se esperen en el punto medio, es decir, ambos hilos ejecutarán sus primeros sysout y se esperarán mutuamente para ejecutar sus segundos sysout.
- **ejercicio 4:** Generar un interbloqueo de dos o más hilos utilizando semáforos.
- **ejercicio 5:** Dadas las siguientes clases:

```
public class Contador {  
    public static int valor = 0;  
}  
  
public class Incrementador extends Thread {  
    public Incrementador() {  
        super();  
    }  
    @Override  
    public void run() {  
        for (int i = 0; i < 20; i++) {  
            Contador.valor++;  
        }  
    }  
}  
  
public class Main {  
    public static void main(String[] args) {  
        Thread inc1 = new Incrementador();  
        Thread inc2 = new Incrementador();  
        inc1.start();  
        inc2.start();  
        while ( inc1.isAlive() || inc2.isAlive() ) {  
            continue;  
        }  
        System.out.println(Contador.valor);  
    }  
}
```

Colocar semáforos (modificar el código necesario) para que el valor final del atributo compartido *valor* de la clase *Contador* sea siempre 40.

- **ejercicio 6:** Tenemos un sistema con 5 terminales y 3 impresoras, En cada terminal se ejecuta un proceso que utiliza una de las 3 impresoras (siendo estas 3 impresoras recursos comunes a las 5 terminales). Programar los procesos (hilos) utilizando semáforos para que el acceso a la impresora sea en exclusión mutua pero pudiendo haber varios procesos a la vez en las impresoras, cada uno en una distinta. Indicar en que impresora se está imprimiendo el documento.
- **ejercicio 7:** Imaginemos una ruta con dos carriles (mano y contramano) y en un determinado lugar un puente de un solo carril, por lo tanto, solo un auto a la vez pasarán por el puente (descartar la posibilidad si dos autos vienen en el mismo sentido). Hacer el programa para que 5 autos crucen el puente.
- **ejercicio 8:** Ahora resulta que la ruta cruza dos puentes, programar el comportamiento de los autos en sentido Norte/Sur y Sur/Norte para que crucen ambos puentes.



- **ejercicio 9:** Una tribu de salvajes cena en comunidad en una gran olla que contiene 3 chuletas. Cuando un salvaje quiere comer el mismo se acerca a la olla (siempre que no haya otro) y toma una chuleta a menos que la misma este vacía; si la olla está vacía el salvaje despierta al cocinero y espera a que este llene la olla. Programar los hilos de salvaje y cocinero para simular la cena.

*Nota: Se permite que el cocinero espere continuamente (while true).*